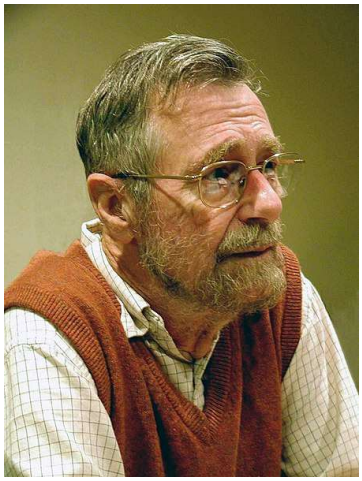
Kishore Muthu

Edsger W. Dijkstra



Personal Life

Edsger W. Dijkstra was born in Rotterdam, Netherlands in 1930 to a chemist father and mathematician mother. Dijkstra first studied theoretical physics at Leiden University but went on to pursue a career in computer science. While at Leiden University, Dijkstra was offered a job at the Mathematical Centre in Amsterdam and officially became the Netherlands' first "programmer" in 1952. In 1957, he would marry Maria C. Debets and in compliance with Dutch marriage rites he was asked to state his profession and his response was that he was as a programmer. Upon hearing this response, the authorities of the town rejected this answer on the grounds that no such job existed, as the field computer science was largely unknown at the time.

Dijkstra continued working at the Mathematical Centre until 1962. Following this, he held a professorship at the Eindhoven University of Technology in the Netherlands and while there he undertook research for the Burroughs Corporation, an American computer manufacturer. A change in direction of the Burroughs Corporation compounded with little support for his work at his university compelled Dijkstra obtain a position at the University of Texas until his retirement in 2000. In early 2002, Dijkstra discovered that he was terminally ill and returned to the Netherlands with his wife and he would pass away soon after.

Contributions

Dijkstra was a pioneer in the field of computer science, and this can be demonstrated through his numerous contributions. His most famous contribution is that of the shortest path algorithm or now know as Dijkstra's algorithm which has now proved to be indispensable for GPS navigation systems. Dijkstra was a big fan of the programming language AGOL 60 and the first compiler for the language was implemented by himself and his partner, Jaap Zonneveld. One of the novel features of this compiler was recursion and this implementation of recursive procedures is now a standard technique in compiler writing.

While working at the Eindhoven University Of Technology, Dijkstra would develop an operating system known as THE multiprogramming system, which had an innovative functional structure. THE was a parallel program which involves multiple components of a program running simultaneously. The components of a parallel

program must be synchronized and must not become deadlocked. In the early 1960s, Dijkstra would identify a crucial problem with parallel programs which he named the “mutual exclusion” problem.

He would put forth a solution to the mutual exclusion problem and a solution for the deadlock problem called the “banker’s algorithm”.

Despite having programmed extensively in machine code, Dijkstra held a low view of the “goto” statement. In 1968, Dijkstra sent a two-page letter addressed to the editor of the Communications of the ACM, which was called “Go To Statement Considered Harmful, and he would write the 1968 article “A Case against the GO TO Statement” which was a major step towards phasing out the goto statement and introducing new control constructs such as the while loop. As a result of his actions, every programmer is now aware of the problems regarding the use of the goto statement and even newer programming languages like Java, released in 1996, don’t have the goto statement.

Another area which was heavily influenced by Dijkstra is the field of distributed computing. It was in 1974 that Dijkstra would introduce the concept of “self-stabilization” in a 2-page article. The article described protocols as to how a system of communicating machines may correct any faults that arises in any one of the machines in the system. A main area where a concept like this is applied in is the world wide web. The significance of this paper was not recognised until 1983 and in 2002 the paper won an award called the “Edsger W. Dijkstra Prize in Distributed Computing”.

An indirect contribution of Dijkstra is in the field of software engineering. In his 1962 paper, “Some Meditations on Advanced Programming”, Dijkstra raised the issue of program correctness and the need for working software. At the time of this paper, the advancement in the power of computers led to increasingly complex and unreliable software which was known as the “software crisis”. Dijkstra spent many years arguing for the recognition of the software crisis as an urgent problem and argued that software systems should be built on sound design principles and program correctness.

Legacy

Through his numerous contributions to the field of computer science, Edsger Dijkstra has left behind an astounding legacy. For example, the shortest path algorithm also known as Dijkstra’s algorithm is taught to all students studying computer science. Dijkstra’s operating system known was “THE multiprogramming system” has influenced many subsequent operating systems primarily due to its layered design.

Dijkstra’s views on program correctness and software design heavily influenced future aspects of structuring and designing programs and contributed to the rise of object-oriented programming. His work also influenced the design of future programming languages and in the development of future programming courses.

Through his countless contributions and his many awards, I believe that Edsger Dijkstra has had a profound and lasting impact in the field of computer science.

References

Dijkstra's Wikipedia :

https://en.wikipedia.org/wiki/Edsger_W._Dijkstra

Dijkstra's Biography :

https://inference-review.com/article/the-man-who-carried-computer-science-on-his-shoulders

IEEE Computer Society :

https://www.computer.org/profiles/edsger-dijkstra

https://history.computer.org/pioneers/dijkstra.html

A.M. Turning Award :

https://amturing.acm.org/award_winners/dijkstra_1053701.cfm